Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Original) A system for providing out-of-band notification of service changes, comprising:

a cluster framework into a layered architecture, comprising:

an application layer comprising at least one of applications and middleware supporting the applications;

a database instance resource group interoperating with the application layer and comprising a database instance providing services; and

a monitor associated with the database instance resource group and exporting an out-of-band interface to the database instance resource group; and

a notification mechanism generating an UP service notification from the cluster framework upon service availability and generating a DOWN service notification from the cluster framework upon service non-availability.

2. (Original) A system according to Claim 1, further comprising:

a planned operation interface incorporated into the application layer; and

the notification mechanism generating a COMING UP service notification responsive to an instruction received through the planned operation interface and generating a GOING DOWN service notification responsive to a further instruction received through the planned operation interface.

3. (Original) A system according to Claim 1, further comprising:

a global services daemon interfaced to the database instance resource group; and

the notification mechanism generating a DOWN service notification for the services on a failed database instance; generating a COMING UP service notification from the global services daemon responsive to a recovering database instance and generating an UP service notification from the global services daemon responsive to a recovered database instance.

4. (Original) A system according to Claim 1, further comprising:

at least one of a remote procedure call interface and an event interface provided to the database instance resource group.

5. (Original) A system according to Claim 1, further comprising:

a resilient set of cluster frameworks comprising an active node and one or more standby nodes.

6. (Original) A system according to Claim 5, wherein the resilient cluster framework executes a node failover to the active node.

7. (Original) A system according to Claim 5, wherein the resilient cluster framework executes a node failover to one such standby node.

8. (Original) A system according to Claim 1, further comprising:

a non-resilient set of cluster frameworks comprising an active node.

9. (Original) A system according to Claim 1, wherein the resilient cluster framework terminates service on a failed node responsive to a DOWN service notification.

10. (Original) A system according to Claim 1, wherein the resilient cluster framework resumes service on a recovered node responsive to an UP service notification.

11. (Original) A system according to Claim 1, wherein the resilient cluster framework effects a switchover to a standby node responsive to a COMING UP service notification.

12. (Original) A system according to Claim 1, wherein the application layer pre-connects to a standby node responsive to one of a COMING UP service notification and an UP service notification.

13. (Currently amended) A **computer-implemented** method for providing out-of-band notification of service changes, comprising:

configuring ~~structuring~~ a cluster framework into a **set of layers, wherein said set of layers comprise:** ~~layered architecture, comprising:~~

an application layer comprising at least one of applications and middleware supporting the applications;

a database instance resource group interoperating with the application layer and comprising a database instance providing services; and

a monitor associated with the database instance resource group and exporting an out-of-band interface to the database instance resource group;

generating an UP service notification from the cluster framework upon service availability; and

generating a DOWN service notification from the cluster framework upon service non-availability.

14. (Currently amended) A **computer-implemented** method according to Claim 13, further comprising:

incorporating a planned operation interface into the application layer;

generating a COMING UP service notification responsive to an instruction received through the planned operation interface; and

generating a GOING DOWN service notification responsive to a further 6 instruction received through the planned operation interface.

15. (Currently amended) A **computer-implemented** method according to Claim 13, further comprising:

providing a global services daemon interfaced to the database instance resource group;

generating a DOWN service notification for the services on a failed database instance;

generating a COMING UP service notification from the global services daemon responsive to a recovering database instance; and

generating an UP service notification from the global services daemon responsive to a recovered database instance.

16. (Currently amended) A **computer-implemented** method according to Claim 13, further comprising:

providing at least one of a remote procedure call interface and an event interface to the database instance resource group.

17. (Currently amended) A **computer-implemented** method according to Claim 13, further comprising:

configuring a resilient set of cluster frameworks comprising an active node and one or more standby nodes.

18. (Currently amended) A **computer-implemented** method according to Claim 17, further comprising:

executing a node failover to the active node.

19. (Currently amended) A **computer-implemented** method according to Claim 17, further comprising:

executing a node failover to one such standby node.

20. (Currently amended) A **computer-implemented** method according to Claim 13, further comprising:

configuring a non-resilient set of cluster frameworks comprising an active node.

21. (Currently amended) A **computer-implemented** method according to Claim 13, further comprising:

terminating service on a failed node responsive to a DOWN service notification.

22. (Currently amended) A **computer-implemented** method according to Claim 13, further comprising:

resuming service on a recovered node responsive to an UP service notification.

23. (Currently amended) A **computer-implemented** method according to Claim 13, further comprising:

effecting a switchover to a standby node responsive to a COMING UP service notification.

24. (Currently amended) A **computer-implemented** method according to Claim 13, further comprising:

pre-connecting to a standby node responsive to one of a COMING UP service notification and an UP service notification.

25. (Currently amended) A computer-readable storage medium holding ~~code~~ **instructions for causing a processor to execute** ~~for performing~~ the **computer-implemented** method according to Claim 13.

26. (Original) A system for communicating service change events in a cluster 2 framework environment, comprising:

a plurality of service change events for communication between a plurality of nodes, comprising:

an UP service change event;

a DOWN service change event;

a COMING UP service change event; and

a GOING DOWN service change event;

a remote procedure call interface from a database instance in a database stack executing on one such node; and

a notification mechanism publishing at least one such service change event from the database instance.

27. (Original) A system according to Claim 26, further comprising:

a further notification mechanism receiving the one such service change event at one or more other nodes.

28. (Original) A system according to Claim 26, further comprising:

a cluster service within the database stack.

29. (Original) A system according to Claim 26, further comprising:

a planned interface within the database stack.

30. (Original) A system according to Claim 26, further comprising:

a global services daemon with listener within the database stack.

31. (Original) A system according to Claim 26, further comprising:

a cluster service processing a multiple instance failover from the one such node to one or more other nodes.

32. (Original) A system according to Claim 26, further comprising:

a cluster service processing a single instance failover to the one such node.

33. (Original) A system according to Claim 26, further comprising:

a cluster service processing a switchover from the one such node to one or more other nodes.

34. (Currently amended) A **computer-implemented** method for communicating service change events in a cluster framework environment, comprising:

defining a plurality of service change events for communication between a plurality of nodes, comprising:

an UP service change event;

a DOWN service change event;

a COMING UP service change event; and

a GOING DOWN service change event;

exporting a remote procedure call interface from a database instance in a database stack executing on one such node; and

generating a notification publishing at least one such service change event from the database instance.

35. (Currently amended) A **computer-implemented** method according to Claim 34, further

comprising:

receiving the one such service change event at one or more other nodes.

36. (Currently amended) A **computer-implemented** method according to Claim 34, further comprising:

providing a cluster service within the database stack.

37. (Currently amended) A **computer-implemented** method according to Claim 34, further comprising:

providing a planned interface within the database stack.

38. (Currently amended) A **computer-implemented** method according to Claim 34, further comprising:

providing a global services daemon with listener within the database stack.

39. (Currently amended) A **computer-implemented** method according to Claim 34, further comprising:

processing a multiple instance failover from the one such node to one or more other nodes.

40. (Currently amended) A **computer-implemented** method according to Claim 34, further comprising:

processing a single instance failover to the one such node.

41. (Currently amended) A **computer-implemented** method according to Claim 34, further comprising:

processing a switchover from the one such node to one or more other nodes.

42. (Currently amended) A computer-readable storage medium holding ~~code~~ **instructions** for **causing a processor to execute** ~~performing~~ the **computer-implemented** method according to Claim 34.

43. (Currently amended) A **computer-implemented** method for detecting a failure of a first process, the method comprising the steps of:

establishing a first connection between said first process and a second process;

monitoring status of said first process to determine whether said first process has failed; and

in response to determining that said first process has failed, notifying said second process that said first process has failed;

wherein a second connection, that is different from said first connection, is used to notify said second process of said failure of said first process failure.

44. (Currently amended) A **computer-implemented** method according to Claim 43, wherein:

the step of establishing a first connection between said first process and a second process includes the step of establishing a first connection between an application server and a database instance;

the step of monitoring includes the step of monitoring status of said database instance; and

the step of notifying said second process that said first process has failed includes the step of causing an out-of-band break to be sent to said application server.

45. (Canceled)

46. (Canceled)